Using Named Entities to Discover Heterogeneous Events on Twitter

Amosse Edouard

Université Côte dAzur, Inria, CNRS, I3S, France
amosse.edouard@unice.fr

Abstract. Social media sites such as Twitter[1] and Facebook[2] have emerged as powerful means of communication that allow people to exchange information about their daily activities, latest news or real-world events. Aside social interactions among users, social medias are expected to provide added value services in a variety of domains (e.g sentiment analysis, trend analysis and event detection). Detecting events on social medias poses new challenges due to the sparsity and the informal nature of social media posts. One of the main challenges in detecting events in social media is to differentiate event and non event messages. To face this challenge, we propose to take advantage from the knowledge that can be extracted from the Linked Opened Data (e.g. DBpedia[3]) to enrich the short textual messages with contextual information brought by the presence of named entities. We evaluate our approach on two gold-standard datasets and the preliminary results show that exploiting the ontological categories of the named entities has a positive impact on the classification output.

Keywords: Event Detection, NLP, Supervised Classification, Named Entities

1 Problem Statement and Motivation

The analysis of social media streams, particularly Twitter, has gained a lot of interest, both within the academic and business communities. The capability to understand and analyse the stream of messages on Twitter is an effective way to monitor what people think [11], what trending topics are emerging [19], and which main events are affecting people's lives. For this reason, several automated ways to track and categorise *events* on Twitter have been proposed in the literature. However, the sheer amount of information contained in Twitter makes it a challenge compared to other source of information such as media news. There are two main reasons for this. One is the larger and real time amount of information compared to media news. Second, the characteristics of tweets (e.g. limited to

[1] http://twitter.com/
[2] http://facebook.com/
[3] http://wiki.dbpedia.org/about

140 characters long) and specific language. More importantly, contrary to media news, not all information from Twitter is related to events [10].

In the recent years, detecting events by analyzing tweets have been widely researched in the field of information retrieval. Although several approaches have focused on the detection and the analysis of large-scale events from Twitter, most of them are concentrated in detecting events of particular types [2, 8, 20]. Such approaches are highly dependant of the target events and generally rely on specific keywords to filter event-related tweets. Their strong connection to the target event makes them unsuitable for detecting events at Twitter's scale.

The different definition carried out within the Topic Detection and Tracking (TDT) and Natural Language Processing (NLP) communities do not seem to capture the particularities of events in social media contents. Commonly, an event is defined as "Something that happens at certain time in certain place and the unavoidable consequences" [1]. Based on this definition and while observing posts on Twitter, we can observe a strong connection between events, and the cited Named Entities (NE) that are involved. In the NLP community, NEs are defined as words or sequence of words on a natural language text that are reference to objects, for example persons, places and organisations. Thus, the presence of the NEs in the content of the tweets might be a good indicator that the tweet is related to event.

In this paper, we experiment with ways of detecting and classifying real-world events on Twitter by relying on the knowledge that can be extracted from knowledge bases such as DBpedia, to enrich tweets with contextual information brought by the presence of named entities involved in the events.

2 Related Works

Existing works on event detection on Twitter can be classified into two main categories:

1. Close-domain : interested in detecting known event type (e.g. earthquake [20], flu [4, 19] or incidents [2])
2. Open-Domain : interested in detecting unknown event

Although, our goal is to identify heterogeneous events (i.e. event of unknown types), approaches that target a particular event types are useful to stress the challenges in detecting events on tweets. This, in the reminder of this section, we review existing works in both categories.

2.1 Close-Domain or Known Event Type

Close-domain works mainly focus on monitoring tweets for detecting known events such as earthquake [20], incidents [2, 3, 21] or social activities [9]. Usually, a set of keywords related to the target event is used to query the Twitter stream for related tweets. Thus, messages containing one or many of such keywords are considered as event-related or ignored otherwise.

Sakaki et al. [20] describe an approach for detecting earthquake by monitoring Twitter messages. The keywords earthquake and shake are used to retrieve relevant tweets on the Twitter stream. To eliminate off-context tweets (e.g. tweets containing 'shake hand'), SVM is used to train a binary classifier using event-related features.

Attardi et al. use discriminative word embeddings as continuous features for training an SVM classifier in the aim of separating tweets related to natural disasters from the others using words related or indicative of disasters as lexicon.

Twitter is also used to extract additional information on existing events. For example, Ritterman et al. [19] use successfully Twitter to predict swine flu pandemic in 2009. Achrekar et al. [4] use tweets related to flu as early indicators of influenza-like illness.
Abel et al. [2, 3] use semantic linking to filter relevant information from tweets about reported incidents in an emergency broadcasting service. Keywords related to the reported incident are used to extract relevant tweets. Semantic linking is used to find particular information pieces in the relevant tweets.

In [21], NEs are experimented in the task of generalizing a binary model trained for detecting incidents in a city to a different city. The authors show that the replacement of NEs helps in holding the performance of a supervised model when using different corpora as training and test set. However, the scope of this work was limited to incidents and only named entities of type location are used. Moreover, their datasets were collected using keywords related to incident (e.g. crash, traffic or explosion) and limited to two cities in the U.S.

Works that target a particular event type are focus on keywords to identify tweets that are related to events. Also, the event type is used to create event patterns for detecting fine-grained topics [14, 22, 24] or labels for training Machine Learning classifiers [5, 20]. Due to their strong connection to the target event, such approaches cannot be applied to event detection at Twitter's scale unless one knows the keywords corresponding to each event which is not trivial.

2.2 Open-Domain or Unknown Event Type

Usually, two main techniques are used to detect open-domain events on Twitter : Document-Pivot or Feature-Pivot techniques [6]. In the former, documents are clustered on the basis of their textual similarity, while the latter monitors bursty terms (terms that are observed at an unusual rate) in a collection of documents where a bursty term is considered as indicator of an event.

Petrovic et al.[16] address the task of First Story Detection by analysing solely the contents of tweets. The presented approach is based on local sensitive hashing, a randomized technique that reduces time needed to find a nearest neighbor in a vector space. Each new tweet is assigned to the thread that contains the most similar tweets where similarity is built using cosine similarity score. The growth rate of thread is used to eliminated non-event related threads, such that threads that grow fastest are considered as event-related and conversely.

Ritter et al. [18] modeled events on Twitter as a 4-tuple representation including NEs, temporal expressions, event phrases and event type. NEs and temporal expressions are extracted using Twitter specific tools [17] while event phrases are extracted using a supervised method. The system recognizes event triggers as a sequence labeling task using Conditional Random Field; then an unsupervised approach is used to classify the events into topics. In addition, the authors consider the association strength between named entities and temporal expressions to decide whether or not a tweet is related to event. This assumption restricts the approach to tweets that explicitly contain temporal expressions and NEs.

In [25], tweets are grouped according to a time-window and BOW is used as document vector. The most frequent terms during the time-window are considered as bursty topics and considered as indicator of events. Bursty terms in the whole historical data are estimated by observing the distribution of terms in the considered time-window following an optimization problem.

Most existing works on open-domain event detection in Twitter are based on the velocity of which the clusters are growing; thus, clusters that grow fastest are considered as event-related [18, 25] or simply discarded otherwise. Although this assumption helps in discovering large-scale events [15], it is less suitable for events with a small audience on Twitter. In our approach, instead of creating event clusters on the whole Twitter data; we propose to separate event and non-event tweets in a separated task. Second, we create event clusters using only tweets that are related to events. By doing this, we minimize the risk of false alarms due to non-event related tweets in the clusters and second, we reduce the time needed to create the clusters.

3 Research Questions

Our overall research question is: how to detect open-domain events by monitoring Twitter messages? What is the best approach to build a event detection model that can hold good performance as time passes? To answer this, the following research questions will be investigated:

RQ 1: Is it possible to use supervised classification to separate event-related and not event-related tweets? What is the impact of in-domain data on classification, especially on overfitting?

RQ 2: How information contained on the LOD (as knowledge about NEs) can contribute to this task in order to mitigate the effects of overfitting on in-domain data?

RQ 3: How can we cluster/categorize events into finer-grained topics?

4 Hypothesis

Based on the research questions listed in Section 3, we make the following hypotheses:

H 1: Separating event-related and non-event related tweets can contribute in reduction the computational time of event detection algorithms

H 2: The presence of NEs in the content of a post is a good indicator that it is related to an event.

H 3: Replacing NEs in tweets by their corresponding category in an ontology can reduce the negative effect of overfitting on a classifier.

5 Our Approach

We now describe our approach for detecting heterogeneous events on Twitter i.e. unknown event types. Based on the research questions and the hypothesis, we present our approach in three steps:

1. First, we separate event-related tweets from the rest of the micro-posts by combining techniques from Machine Learning (ML), NLP and LOD.
2. Second, we classify the tweets that are related to events into coarse-level categories as described in the TDT manual [1] including: Science, Armed Conflicts, Politics, Economy, Culture, Sports, Accidents and Miscellaneous.
3. Third, we propose to cluster the tweets in each category into finer-grained topics by grouping similar tweets using a feature-pivot technique.

In the remainder of this section, we detail how we plan to carry out our work on each step.

5.1 Identifying Event-Related Tweets

In previous works [18, 21], events are typically defined according to time, space and agents involved such as locations, persons or organisations, denoted as Named Entities (NE). For the first and second step, we propose to build a classification model based on semantic abstraction on the NEs. The semantic abstraction consists in replacing the NEs cited in tweets by their ontological categories(e.g. types in DBpedia) and use the modified content to extract features for training a supervised model. First, we link the NE mentions in tweets to resources in a knowledge base (e.g. DBpedia); second, we replace the NEs by their category in the ontology, and third, we create a feature vector with the modified content. Finally, we use the feature vector to train the supervised model.

Named Entity Recognition, Linking and Replacement We utilize the NERD-ML [23] tool to perform Named Entity Recognition (NER) and entity linking. Our choice of NERD-ML is motivated by the work in [7], which has found that NERD-ML performs better on Twitter data than other Twitter-specific NLP tools such as Tweet NLP [17]. We use the SPARQL[4] query language to retrieve the categories of the NE in the knowledge base. Additionally, we sort the output of the query according to the hierarchy of the ontology. We

[4] `https://www.w3.org/TR/rdf-sparql-query/`

experiment two NER techniques, namely generic and specific replacement. In the former, the NEs are replaced by their most generic category;[5] in the latter we replace the NEs by their most specific category.[6] Two example outputs of the Entity Replacement module are reported in Table 1. The rationale behind the replacement of entity mentions with their type is to generalise over single mentions, thus avoiding overfitting in supervised settings.

Original Tweets	**Generic Categories**	**Specific Categories**
Cambodia's ex-King Norodom Sihanouk dead at 89 http://q.gs/2IvJk #FollowBack	[Place] ex-king [Person] die at [number]	[Country] ex-king [Royalty] die at [number]
Amy Winehouse, 27, dies at her London flat http://bit.ly/nD9dy2 #amyWinehouse	[Person], [number], die at her [Place] flat [Person]	[Person], [number], die at her [Settlement] flat [Person]

Table 1. Examples of the output of the Entity Replacement module on tweets. For simplicity, in this table we only show categories in DBpedia ontology.

Classification Approach In order to separate event tweets and non events tweets and to associate a coarse-level category [1, 12] to event-related tweets, we use a supervised method. We consider two ways to build the supervised model: (1) A binary classifier that classifies the tweets into events and non events provides the input to second model to associate an event category to the tweets. (2) A single multi-class classification: A model trained on 9 classes, including the 8 event categories plus a non event-related class. We plan to experiment the two approaches and select the one that gives the best results.

5.2 Extracting Event Topics

The third step of our approach is topic detection. Giving a set of tweets label as related to events in the previous tasks, the goal is to detect fine-grained event topic (e.g. The death of Amy Winehouse). Contrary to the first step, it is inconceivable to build a supervised model for this task since the possible event topics in Twitter are unknown in advance [18]. Instead, we propose to build an unsupervised model exploiting the event categories output by the supervised model.

We create topic clusters in each event category by grouping together similar tweets. Due to the sparsity of tweets, we propose to use a feature-pivot techniques instead a document-pivot techniques [6]. Instead of considering each

[5] `i.e. The last category in the hierarchy (excluding the Thing class)`
[6] `i.e. The first category in the hierarchy`

word in a tweet as a bursty candidate, we reduce the feature space by considering relationship between NEs and event phrases (i.e. action verbs) as in [?]. Furthermore, using the relationship between NEs and event phrases is useful to separate events sharing a common type into specific event topics (e.g. an accident that occurs in different places). Also, we use Wordnet [13] to extract the synsets for the event phrases in order to group similar clusters.

Finally, following the state of the art approaches, emergent event topics are identified by monitoring the growth of each clusters. Since this task is connected to the classification task, we are sure that the clusters are related to events; thus, we use the growth rate of the clusters to sort the events according to their popularity in Twitter such that clusters that grow fastest are the most popular.

6 Evaluation Plan

In this section, we defined the evaluation strategies for our research questions. Since we are interested in detecting events on Twitter, our approaches are evaluated on two gold-standard datasets. In the reminder of this section, we will present the characteristics of each dataset as well as the valuation strategies for each research question.

6.1 Datasets

For our experiments, we choose two gold-standard corpus of tweets collected over two distinct periods and cover very different topics. Both datasets are manually annotated and tweets related to events are annotated either with a coarse-level event category (e.g. Culture) and fine-level event type (e.g. The death of Amy Winehouse).

The Events 2012 Corpus [12] A total of 120 million tweets were collected from October to November 2012 from the Twitter streaming API,[7] of which 159,952 tweets were labeled as event-related. 506 event types were gathered from the Wikipedia Current Event Portal, and Amazon Mechanical Turk was used to annotate each tweet with one of such events. Besides, each event was also associated with an event category following the TDT annotation manual [1]. Events covered by this dataset include for example the US presidential debate between Barack Obama and Mitt Romney, the US presidential election results or the Chemistry Nobel prize. According to Twitter policies, only tweet identifiers can be released. Therefore, we use these identifiers to download the original contents of the tweets from the Twitter platform. After removing duplicated tweets and those that are no-longer available, we are left with $\sim$ 92 million tweets from which 42,334 tweets related to events.

[7] `https://dev.twitter.com/streaming/overview`

First Story Detection Corpus (FSD) [16] This corpus consists of a set of 50 million Twitter messages collected from July 2011 until September 2011 using the Twitter API. Two human annotators annotated the tweets related to events with one out of 27 event types extracted from the Wikipedia Current Event Portal; agreements between the annotators using Cohen's kappa was 0.65. In total, 3,035 tweets were labeled as related to events and annotated with a corresponding event topic (e.g. 'death of Amy Winehouse', 'earthquake in Virginia' or 'plane crash of the Russian hockey team'). After removing tweets that are no more available, we are left with ∼31 million tweets from which 2,250 are related to events.

Contrary to the Event 2012 corpus, the events in the FSD corpus are not associated with event categories. Therefore, in order to merge the two corpora in a single dataset for our experiments, we extended the FSD corpus by labelling each event topic with one of the event categories of the Event 2012 corpus [1, 12]. The task was manually performed by three annotators: the labels were first assigned independently, and then adjudicated by majority vote in case of disagreements.[8] Agreement between the three annotators, measured using Krippendorffs alpha coefficient, was (alpha = 0.758). Table 2 shows the number of tweets in each corpus divided into categories.

Event Category	**Event 2012**	**FSD**
Arts	2589	710
Attacks	7079	56
Politics	16383	58
Sports	8812	0
Economy	2881	342
Science	1537	296
Accidents	2479	778
Miscellaneous	574	10
Total	42334	2250

Table 2. Tweets in each event category

6.2 Evaluation Strategies for the First and Second Research Question

To demonstrate the importance of NLP in detecting open-domain events on Twitter, we compare our NLP-based approach against a baseline which does not make use of NLP nor LOD. On the other hand, our evaluation aims to prove that it is feasible to use supervised machine learning to separate event related tweets and non event related tweets.

[8] The Web interface used for annotation is available at `http://www.i3s.unice.fr/~edouard/events/agreements.html`

Since both corpora contain much more non-event related than event related tweets, resulting in a very skewed class distribution, we reduced the number of negative instances by randomly selecting a sample of non event-related tweets. The final amount of tweets in the two datasets is reported in Table 3.

We consider two evaluation settings namely, Setting 1 and Setting 2. In Setting 1, we evaluate the model using 10-fold cross validation only with the Event 2012 corpus. In Setting 2, we use the Event 2012 as training set and the FSD corpus as test set. Our focus is on Setting 2 (i.e. train the model on a corpus and test it on an other); however, Setting 1 is useful to understand the effect of only in-domain data on the output of supervised method.

	Event-related	**Non event-related**	**Total**
Event 2012	42,334	48,239	90,573
FSD	2,250	3,040	5,290

Table 3. Total number of tweets per dataset

6.3 Evaluation Strategies for the Third Research Question

To evaluate our approach for detecting event topic, we will use the event topics described in the datasets we described in Section 6.1 as our ground truth. Our evaluation strategy is two fold: (1) We evaluate the ability to determine the correct event topics and (2) we compare the summary of the topics with the summary of each events in the datasets. For each evaluation strategy, we will use human annotators to evaluate measure the similarity between the topics associated to each tweet by our approach and the correct topic in the datasets. Using the human annotations, we can compute the evaluation metrics such precision and recall.

7 Preliminary Results

We have already made some experiments to evaluate our approach for separating tweets related to events from the rest of tweets. We compare the results of our approach against a simple baseline which does not make use of NLP nor LOD on the basis of precision, recall and f-measure. We train two models according using Setting 1 and Setting 2; the results are depicted in Tables 4 and 5.

7.1 Experimental Setup

Before training the classifiers, we further clean up the datasets. We remove any Twitter-specific features such as URLs, user mentions, emoticons and duplicate tweets.[9] We also perform common prepossessing tasks such as stop words removal and stemming. We employ character sequence n-gram features [9] and model the feature vector using bag-of-words weighted by TF-IDF.

We consider the NER techniques as described in Section 5 for training a binary classifier to separate event and non event tweets. For each setting, we select all tweets related to events as positive instances and randomly selected an equivalent number of non event-related tweets as negative instances. Finally, we use the Weka[10] tool to train a Naive Bayes and an SVM classifier.

7.2 Results

Table 4 depicts the evaluation of our approach as well as the baseline, on the Setting 1 (i.e train and test on the same dataset). In setting 1, the baseline outperforms our method in precision and recall for both NB and SVM classifier. Our results are similar to those obtained by Ilina et al. [9] who found that training and testing on the same datasets obtain higher performance than training and testing on different datasets. A plausible reason for that is overfitting due to the similarity between training and testing instances.

We also conduct the experiments with Setting 2 (i.e we train on the Event 2012 dataset and test on the FSD dataset) and reports the results in Table 5. With this configuration, the best performing method is obtained when the NEs are replaced by their most generic class in the DBpedia ontology outperforming the baseline. As expected, both methods obtain lower performance regarding Setting 1. Nevertheless, our method yields a drop of 0.028, the baseline yields a drop of 0.167 in f-measure. These preliminary results show that using the ontological categories of the NEs, we mitigate the impact of overfitting on the output of our supervised model.

	Naive Bayes			SVM		
Approach	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
dbp:generic.	0.897	0.896	0.896	0.906	0.903	0.904
dbp:specific.	0.871	0.869	0.870	0.887	0.884	0.885
Baseline	0.919	0.918	0.918	**0.951**	**0.949**	**0.950**

Table 4. Evaluation on setting 1: A NB and an SVM classifier are trained and tested with tweets from the same dataset using 10-fold cross-validation.

8 Discussion and Future Work

This paper presents a proposal to tackle the problem of detecting open-domain events on Twitter and our preliminary results. The main idea of the proposed

[9] After removing Twitter-specific features and replacing named entities, we remove the duplicates tweets.

[10] http://www.cs.waikato.ac.nz/ml/weka/

	Naive Bayes			SVM		
Approach	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
dbp:specific.	0.811	0.809	0.810	**0.879**	**0.875**	**0.876**
dbp:generic.	0.810	0.809	0.809	0.873	0.873	0.873
Baseline	0.783	0.784	0.783	0.789	0.739	0.763

Table 5. Evaluation on setting 2: A NB and an SVM classifier are trained with tweets from the Event 2012 dataset and tested with tweets from the FSD dataset.

approach exploits the relation between events and NE but also NLP and LOD techniques to build a supervised method to separate event and non event tweets. The output of the supervised modeled is used as input for a clustering approach in which event categories are exploited to detect finer-grained event topics. We found that the replacement of the NEs in tweets by their associated concepts in the DBpedia ontology has proved to be efficient in reducing the negative effect of overfitting in the output of our model. Our preliminary results show that the proposed approach holds higher precision and recall compared to a baseline when the training and test sets are different. For future works we plan to improve the approach by considering categories from other ontologies such as Yago; we also plan to experiment other classifiers such as Neural Network.

Acknowledgments. The author would like to thanks Elena Cabrio, Sara Tonelli and Nhan Le-Than for helping to develop this research plan, their writing tips and the kind review of the early drafts of this paper. I am also grateful to the anonymous reviewers for their valuable and constructive feedback.

References

1. TDT 2014: Annotation Manual. `https://catalog.ldc.upenn.edu/docs/LDC2006T19/TDT2004V1.2.pdf` (2014), [Online; accessed 03-March-2016]
2. Abel, F., Hauff, C., Houben, G.J., Stronkman, R., Tao, K.: Semantics+ filtering+ search= twitcident. exploring information in social web streams. In: Proceedings of the 23rd ACM conference on Hypertext and social media. pp. 285–294. ACM (2012)
3. Abel, F., Hauff, C., Houben, G.J., Stronkman, R., Tao, K.: Twitcident: fighting fire with information from social web streams. In: Proceedings of the 21st international conference companion on World Wide Web. pp. 305–308. ACM (2012)
4. Achrekar, H., Gandhe, A., Lazarus, R., Yu, S.H., Liu, B.: Predicting flu trends using twitter data. In: Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on. pp. 702–707. IEEE (2011)
5. Anantharam, P., Barnaghi, P., Thirunarayan, K., Sheth, A.: Extracting city traffic events from social streams. ACM Transactions on Intelligent Systems and Technology 9(4) (2014)
6. Atefeh, F., Khreich, W.: A survey of techniques for event detection in twitter. Computational Intelligence 31(1), 132–164 (2015)

7. Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., Bontcheva, K.: Analysis of named entity recognition and linking for tweets. Information Processing & Management 51(2), 32–49 (2015)
8. Earle, P.S., Bowden, D.C., Guy, M.: Twitter earthquake detection: earthquake monitoring in a social world. Annals of Geophysics 54(6) (2012)
9. Ilina, E., Hauff, C., Celik, I., Abel, F., Houben, G.J.: Social event detection on twitter. In: Web Engineering, pp. 169–176. Springer (2012)
10. Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: understanding microblogging usage and communities. In: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis. pp. 56–65. ACM (2007)
11. Kaplan, A.M., Haenlein, M.: Users of the world, unite! the challenges and opportunities of social media. Business horizons 53(1), 59–68 (2010)
12. McMinn, A.J., Moshfeghi, Y., Jose, J.M.: Building a large-scale corpus for evaluating event detection on twitter. In: Proceedings of the 22nd ACM international conference on Conference on information & knowledge management. pp. 409–418. ACM (2013)
13. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM 38(11), 39–41 (1995)
14. Nichols, J., Mahmud, J., Drews, C.: Summarizing sporting events using twitter. In: Proceedings of the 2012 ACM international conference on Intelligent User Interfaces. pp. 189–198. ACM (2012)
15. Osborne, M., Petrovic, S., McCreadie, R., Macdonald, C., Ounis, I.: Bieber no more: First story detection using twitter and wikipedia. In: SIGIR 2012 Workshop on Time-aware Information Access (2012)
16. Petrović, S., Osborne, M., Lavrenko, V.: Streaming first story detection with application to twitter. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. pp. 181–189. Association for Computational Linguistics (2010)
17. Ritter, A., Clark, S., Etzioni, O., et al.: Named entity recognition in tweets: an experimental study. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1524–1534. Association for Computational Linguistics (2011)
18. Ritter, A., Etzioni, O., Clark, S., et al.: Open domain event extraction from twitter. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1104–1112. ACM (2012)
19. Ritterman, J., Osborne, M., Klein, E.: Using prediction markets and twitter to predict a swine flu pandemic. In: 1st international workshop on mining social media. vol. 9, pp. 9–17. ac. uk/miles/papers/swine09. pdf (accessed 26 August 2015) (2009)
20. Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake shakes twitter users: real-time event detection by social sensors. In: Proceedings of the 19th international conference on World wide web. pp. 851–860. ACM (2010)
21. Schulz, A., Janssen, F.: What is good for one city may not be good for another one: Evaluating generalization for tweet classification based on semantic abstraction. In: Proceedings of the Fifth International Conference on Semantics for Smarter Cities-Volume 1280. pp. 53–67. CEUR-WS. org (2014)
22. Tanev, H., Piskorski, J., Atkinson, M.: Real-time news event extraction for global crisis monitoring. In: Natural Language and Information Systems, pp. 207–218. Springer (2008)

23. Van Erp, M., Rizzo, G., Troncy, R.: Learning with the web: Spotting named entities on the intersection of nerd and machine learning. In: # MSM. pp. 27–30. Citeseer (2013)
24. Wang, X., Gerber, M.S., Brown, D.E.: Automatic crime prediction using events extracted from twitter posts. In: Social Computing, Behavioral-Cultural Modeling and Prediction, pp. 231–238. Springer (2012)
25. Xie, W., Zhu, F., Jiang, J., Lim, E.P., Wang, K.: Topicsketch: Real-time bursty topic detection from twitter. In: Data Mining (ICDM), 2013 IEEE 13th International Conference on. pp. 837–846. IEEE (2013)